UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/821,025	04/08/2004	David Shippy	AUS920030973US1	9228

50170 7590 08/09/2007

IBM CORP. (WIP)
c/o WALDER INTELLECTUAL PROPERTY LAW, P.C.
P.O. BOX 832745
RICHARDSON, TX 75083

EXAMINER	
MEONSKE, TONIA L	

ART UNIT	PAPER NUMBER
2181	

MAIL DATE	DELIVERY MODE
08/09/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

MN

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/821,025
Filing Date: April 08, 2004
Appellant(s): SHIPPY, DAVID

Stephen R. Tkacs
P.O. Box 832745
Richardson, TX 75083

For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 2 April 2007 appealing from the Office action mailed 2 November 2006.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

5,900,025	Sollars	5-1999

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1-5, 7-8 and 16-21 are rejected under 35 U.S.C. 102(b) as being anticipated by Sollars (US Pat. No. 5,900,025).

Regarding **independent claim 1**, Sollars discloses *an architected register file system at least configured to utilize a plurality of threads, comprising: a plurality of register files* [see Sollars, Col. 5, lines 21-30], *wherein each register file of the plurality of register files at least corresponds to a respective thread of the plurality of threads* [see Sollars, Col. 5, lines 24-30; Examiner's note: In this cite, Sollars discloses a multi-dimensional register file which would allow for multiple threads to occupy a register file in a subset dimension. This idea is illustrated in Fig. 2a of US Pat. No. 6,081,880 (to Sollars as well) incorporated by reference by Sollars in '025]; *a plurality of Status and Control Registers (SCR)* [see Sollars, Fig. 1, element 20a; Col. 5, lines 31-34], *wherein each SCR corresponds to a respective thread of the plurality of threads* [see Sollars, Col. 2, lines 2-6]; *and a plurality of control bit sets, wherein each control bit set corresponds to at least one SCR* [see Sollars, Fig. 5; Examiner's note: Fig. 5 illustrates an SCR containing control bit sets.], *and wherein each control bit set is at least configured to allow a thread associated with an associated SCR to utilize other register files associated with other threads* [see Sollars, Col. 15, lines 60-66].

Regarding **claim 2**, Sollars discloses *the architected register file system of claim 1, wherein the architected register file system further comprises a decoder, wherein the decoder at least determines desired operations for an instruction* [see Sollars, Col. 2, line 64 to Col. 3, line 1; Examiner's note: It is inherent that the function of a decoder is to determine the operations requested of an instruction.].

Regarding **claim 3**, Sollars discloses *the architected register file system of claim 1, wherein plurality of control bits further comprise a plurality of bit doublets* [see Sollars, Fig. 9A, elements "sl" and "ll"], *wherein a first bit of a bit doublet corresponds to a read function* [see Sollars, Col. 10, lines 57-62, "...of a load operation is to be locked..."], *and wherein a second bit of the bit doublet corresponds to a write function* [see Sollars, Col. 10, lines 51-57, "...of a store operation is to be locked..."].

Regarding **claim 4**, Sollars discloses *the architected register file system of claim 3, wherein the architected register file system further comprises: an address control, wherein the address control at least determines addresses with the plurality of register files* [see Sollars, Col. 5, lines 24-30; Examiner's note: Sollars incorporates US Pat. No. 6,081,880 (to Sollars), which discloses the operation of the operand, register file (Sollars ('025), Fig. 2, element 22a). In Figure 2a of '880, Sollars shows a multidimensional register file wherein element 22' is used to address register files 22a-*. Therefore, by reference, Sollars shows an address control that determines the address of a plurality of register files.]; *and at least one execution unit* [see Sollars, Fig. 2, element 14], *wherein the execution is at least configured to perform the operations of a input instruction within the plurality of register files* [see Sollars, Col. 6, lines 24-34].

Regarding **claim 5**, Sollars discloses *the architected register file system of claim 3, wherein the plurality of bit doublets further comprises that each bit doublet at least corresponds to enabling the use of at least one register file associated with another* *thread* [see Sollars, Col. 10, lines 51-62 (enabling writing and reading to/from a register file); Col. 15, lines 60-66 (accessing other multiple threads)].

Regarding **independent claim 7**, Sollars discloses *a method for utilizing a plurality of register files* [see Sollars, Col. 5, lines 21-30] *with associated SCRs in a multithread system* [see Sollars, Fig. 1, element 20a; Col: 5, lines 31-34], *wherein each register file is at least associated with one thread of a plurality of threads* [see Sollars, Col. 5, lines 24-30; Examiner's note: In this cite, Sollars discloses a multi-dimensional register file which would allow for multiple threads to occupy a register file in a subset dimension. This idea is illustrated in Fig. 2a of US Pat. No. 6,081,880 (to Sollars as well) incorporated by reference by Sollars in '025], *comprising: receiving an instruction for a first thread of the plurality of threads* [see Sollars, Col. 2, line 64 to Col. 3, line 1, "...fetching...instructions for the active threads..."], *wherein the first thread is at least associated with a first SCR* [see Sollars, Col. 2, lines 2-6; Examiner's note: Since the SCR is associated with a threads register file, it is inherently associated with the thread.]; *decoding the instruction to at least determine performance operations* [see Sollars, Col. 2, line 64 to Col. 3, line 1, "...decoding...instructions for the active threads..."]; *determining if the first thread is enabled to at least utilize register files associated with other threads* [see Sollars, Col. 10, lines 57-62]; *and executing the instruction, wherein the step of executing utilizes at least one register file associated with a second thread of the plurality of threads* [see Sollars, Col. 15, lines 60-66].

Regarding **claim 8**, Sollars discloses *the method of claim 7, wherein the step of determining if the first thread is enabled, further comprises measuring logical levels of control bits associated with the first SCR, wherein the control bits comprise a plurality of bit doublets* [see Sollars, Fig. 9A, elements “sl” and “ll”], *and wherein each bit doubled at least corresponds to enabling the use of at least one register file associated with another thread* [see Sollars, Col. 10, lines 51-62; Col. 15, lines 60-66].

Claims 10-15 have been cancelled.

Regarding **claim 16**, Sollars discloses a method for utilizing a plurality of register files in a multithread system, the method comprising:

receiving an instruction for a first thread having an operations code field, a write field, and one or more read fields, wherein the operations code field defines a desired operation for the instruction, wherein the write field defines an address location to which a result of the operation is to be stored, and wherein the at least one read field defines an address location from which data is to be read for the operation (See figure 16A: Generic instruction format);

decoding the instruction (See column 6, lines 16-18: The IFU decodes instructions);

setting a first status and control register associated with the first thread and a second status and control register associated with a second thread based on the decoding of the instruction (Instructions are meant to set status and configure registers accordingly), wherein a first register file is associated with the first thread and a second register file is associated with the second thread; determining whether the first thread is permitted to utilize the second register file associated with the second thread based on at least one of the first status and control register or the second status and control register (See column 16, lines 5-15); and if the first thread is permitted to utilize the second register file, performing the operation utilizing the second register file by writing to or reading from the second register file associated with the second thread (See column 13, lines 25-38: Once an exception is executed upon being encountered).

Regarding **claim 17**, Sollars discloses the method of claim 16, wherein performing the operation comprises: reading data from the second register file based on an address in a read field within the one or more read fields (See column 15, lines 60-66).

Regarding **claim 18**, Sollars discloses the method of claim 16, wherein performing the operation comprises: writing a result of the operation to the second register file based on an address in the write field (See figure 16A: Writing is done according to the instruction).

Claims 19-21 are rejected for reasons similar to claims 16-18.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 6 and 9 are rejected under 35 U.S.C. 103(a) as being unpatentable over Sollars.

Regarding **claim 6**, Sollars discloses the limitations as stated in **claim 5**.

Sollars also discloses *one bit* [of a doublet] *is at least configured to correspond to a read function* [see Sollars, Col. 10, lines 57-62, "...of a load operation is to be locked..."] and *one bit* [of a doublet] *is at least configure to correspond to a write function* [see Sollars, Col. 10, lines 51-57, "...of a store operation is to be locked..."].

Sollars does not explicitly disclose *a logic high or '1'* enabling *the first thread to read from another register file* or *a logic high or '1'* enabling *the first thread to write to another register file.*

However, it would have been obvious to one of ordinary skill in the art at the time of invention to utilize a logic high value to correspond to the enablement of a value stored within a register. At the time of invention, the use of a logic high signal as an indication of an enabled function would have been common knowledge, such as the concept of register flags indicating certain conditions. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to utilize an active-high scheme to enable access to a register file.

Regarding **claim 9**, Sollars discloses the limitations as stated in **claim 8**.

Sollars does not explicitly disclose *determining if any bits are `1` or logic high, wherein the `1` or the logic high enables the first thread to read or write to another register file.*

However, it would have been obvious to one of ordinary skill in the art at the time of invention to utilize a logic high value to correspond to the enablement of a value stored within a register. At the time of invention, the use of a logic high signal as an indication of an enabled function would have been common knowledge, such as the concept of register flags indicating certain conditions. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to utilize an active-high scheme to enable access to a register file.

(10) Response to Argument

Arguments presented by the Appellant begin on page 6 of the Appeal Brief and continue onto page 15.

Appellant in the last paragraph on page 7 of the Appeal Brief argues,

"The Office Action alleges that the primary and secondary control register files of Sollars are equivalent to the plurality of status and control registers and that the

primary and secondary operand register files of Sollars correspond to the plurality of register files in claim 1. Sollars does not teach that each operand register file corresponds to a thread. Rather, Sollàrs teaches that the primary operand register file includes a number of registers for performing the function of storing instruction operands...Sollars does not teach that each operand register file corresponds to a respective thread."

It is noted that while there is no stated direct correspondence of threads to operand registers, it is explicitly stated in Sollars (column 3, lines 45-57 and column 15, lines 57-67) and acknowledged by the Appellant that there is a correspondence of threads to the control registers (See page 8 of the Appeal Brief). It is also stated by Sollars in column 5, lines 7-19 that figure 1 shows all elements are coupled to each other. Upon inspection of figure 1, it can be seen that the control registers and the operand registers are coupled to each other. This coupling is a form of correspondence. Although there does not exist a direct correspondence, an indirect correspondence does exist and therefore meets the limitations of the claims. Therefore the examiner's rejection should be affirmed.

Appellant in the last paragraph on page 8 of the Appeal Brief argues,

"Sollars does not teach or suggest a plurality of control bit sets that are configured to allow a thread associated with an associated status and control register to utilize other register files associated with other threads...the Office Action alleges that the primary and secondary operand register files are

equivalent to the plurality of register files of claim 1. Thus, the Office Action has not shown that Sollars teaches of fairly suggests a plurality of control bit sets that are configured to allow a thread to utilize other operand register files associated with other threads. In fact, Sollars does not even teach that each operand register file corresponds to a thread."

It is noted that on page 9 of the Appeal Brief, Appellant agrees that Sollars teaches control bits for control registers. It is also noted that there is no explicit limitation in claim 1 that recites that there is a control bit for the <u>operand registers</u>. The explicit limitation reads "each control bit set is at least configured to allow a thread associated with an associated SCR to utilize other register files associated with other threads." The term "other register files" is ambiguous and is interpreted to read "other SCR register files," and not interpreted as "other operand register files" as argued by Appellant.

Furthermore, it can also be argued that the coupling of the operand registers and the control registers would "allow a thread associated with an associated SCR to utilize other register files associated with other threads" using the reasoning in the first argument above. Since everything in the system of Sollars is coupled (column 5, lines 7-19), everything in Sollars is necessarily associated to each other. "Associated" is an overly broad term. Accordingly, the limitation "each control bit set is at least configured to allow a thread associated with an associated SCR to utilize other register files associated with other threads." is met by the Sollars reference and therefore, the examiner's rejection should be affirmed.

Appellant in the top of and continuing onto the first full paragraph of page 11 of the Appeal Brief argues,

"Sollars does appear to teach temporarily conferring a privilege for accessing and modifying a set of control registers...Claim 16, like claim 1, does not recite determining whether a first thread has permission to access the control registers of a second thread; rather, the claims recite determining whether a first thread has permission to access an <u>*operand*</u> *register file associated with a second thread."*

However as similarly argued above, the claims do not require "determining whether a first thread has permission to access an <u>operand</u> register file associated with a second thread". The claimed register files may be interpreted to be any type of register files, including the control register files. Sollars has in fact taught "determining whether the first thread has permission to utilize the second register file associated with the second thread..." (column 3, lines 45-57 and column 15, lines 57-67, The control register set of a thread determines whether a thread has permission to utilize (access and modify) a peers' associated register file.). It is also noted that the control bit of claim 1 is a more specific example of the limitations present in claim 16 and thus the control bits of claim 1 would read onto the permission methodology of claim 16. Accordingly, the limitation "determining whether the first thread has permission to utilize the second register file associated with the second thread..." is met by the Sollars reference and therefore, the examiner's rejection should be affirmed.

Appellant on page 12 of the Appeal Brief argues that column 15, lines 60-66 do not teach

"reading data based on an address in a read field of an instruction...[and]...writing a result to a register file associated with a different thread based on an address in a write field of an instruction."

It is noted that column 15, lines 60-66 teaches thread manipulations, specifically accessing and modifying threads. Accessing a thread is done by reading a thread at a specified address from a read field of an instruction. Modifying a thread is writing to a thread at a specified register address indicated by a write field in an instruction. It is also noted that the Appellant does not argue as to how Sollars does not teach such limitations and merely states that the teachings of Sollars, when compared to the patent application of the Appellant "is very different." Accordingly, the limitation "reading data based on an address in a read field of an instruction...[and]...writing a result to a register file associated with a different thread based on an address in a write field of an instruction" is met by the Sollars reference and therefore, the examiner's rejection should be affirmed.

Appellant on page 13 of the Appeal Brief argues that *"Sollars does not teach or suggest that each operand register file corresponds to a thread.."*

However, as noted above, while there is no stated direct correspondence of threads to operand registers, it is explicitly stated in Sollars (column 3, lines 45-57 and column 15, lines 57-67) and acknowledged by the Appellant that there is a correspondence of threads to the control registers (See page 8 of the Appeal Brief). It is also stated by Sollars in column 5, lines 7-19 that figure 1 shows all elements are coupled to each other. Upon inspection of figure 1, it can be seen that the control registers and the operand registers are coupled to each other. This coupling is a form of correspondence. Although there does not exist a direct correspondence, an indirect correspondence does exist and therefore meets the limitations of the claims. Accordingly, the limitation "each operand register file corresponds to a thread" is met by the Sollars reference and therefore, the examiner's rejection should be affirmed.

Appellant on page 13 of the Appeal Brief argues that

"Sollars also fails to teach or suggest a plurality of control bit sets that are configured to allow a thread associated with an associated status and control register to utilize other registers associated with other threads."

However, Sollars has taught a plurality of control bit sets that are configured to allow a thread associated with an associated status and control register to utilize other registers associated with other threads (column 15, lines 60-66, column 3, lines 44-58, The control register sets (or the claimed control bit sets) each determine for each associated thread which other peer thread's are allowed to be accessed and modified.). Accordingly, the limitation "a plurality of control bit sets that are configured to allow a thread associated with an associated status and control register to utilize other registers associated with other threads" is met by the Sollars reference and therefore, the examiner's rejection should be affirmed.

Appellant on page 14 of the Appeal Brief argues that

"The Final Office Action fails to show that Sollars teaches or suggests a plurality of control bit sets that are configured to allow a thread associated with an associated status and control register to utilize other operand register files associated with other threads."

However, in claims 6 and 9, Appellant has not claimed anything about operand register files. So the fact that Sollars may or may not teach or suggest "a plurality of control bit sets that are configured to allow a thread associated with an associated status and control register to utilize other operand register files associated with other threads" is irrelevant. Therefore the examiner's rejection should be affirmed.

Appellant on page 14 of the Appeal Brief argues that

"Given a lack of an suggestion in the cited prior art to provide a bit doublet in a status and control register to separately enable a thread to read from or write to a register file associated with a different thread, a person of ordinary skill in the art would not have found it obvious to modify the prior art to include this feature. The mere fact that a prior art reference can be readily modified does not make the modification unless the prior art suggested the desirability of the modification. The Office Action has not shown that the prior art teaches or suggests a bit doublet in a status and control register for separately enabling a thread to write to a register file associated with a different thread."

However, Sollars has not only suggested, but has in fact explicitly taught a bit doublet in a status and control register (see Sollars, Fig. 9A, elements "sl" and "ll" comprise the bit doublet, also see Col. 10, lines 51-57) for separately enabling, or allowing, a thread to write to a register file associated with a different thread when the sl value is set to indicate that writing (or storing) is allowed (or not locked).). Sollars has not explicitly disclosed *a logic high or '1'* is the actual value that enables *the first thread to write to another register file.* However, the value of sl needed to be configured or set to some value to enable the writing, Sollars was just silent as to which value would have enabled the writing-a high or low value. It would have been obvious to one of ordinary skill in the art at the time of invention to utilize any value, including a logic high value, to correspond to the enablement of a value stored within a register. At the time of invention, the use of a logic high signal as an indication of an enabled function would have been common knowledge, such as the concept of register flags indicating certain conditions. Therefore, it would have been obvious to one of ordinary skill in the art at the time of invention to utilize an active-high scheme to enable access to a register file. Accordingly, the limitation "a bit doublet in a status and control register for separately enabling a thread to write to a register file associated with a different thread" is met by the Sollars reference and by one of ordinary skill in the art at the time the invention was made and therefore, the examiner's rejection should be affirmed.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Tonia L. Meonske 08/06/2007
Tonia L. Meonske

Conferees:

Alford Kindred

/Lynne H Browne/
Lynne H Browne
Appeal Practice Specialist, TQAS
Technology Center 2100